Machine Learning
Lab Assignment 8
Decision Trees

Note: This assignment is for your understanding of application of decision trees for classification and regression.

Question 1.

Classification: Consider the CarEvaluation dataset (Available from: `https://archive.ics.uci.edu/dataset/19/car+evaluation`) and implement the following.

- Grow the decision tree using the entropy measure (ID3 algorithm) or gini index (CART algorithm). Decision trees tend to be overly complex and do not generalize well to the data and thus, overfit. To overcome this, Stopping criteria can be used such as setting the maximum depth of the tree T. Use cross validation to find the value of depth of the tree, T that gives best performance. Plot average accuracy vs. T. Plot (Visualize) the decision tree constructed on the training data.
- **Ensemble methods to overcome overfitting:**
  - **Bagging:** for the value of T identified above, perform bagging using B bootstrapped datasets (You may consider B in the range of 50 to 500 with intervals of 50). Use cross validation to fix B. Plot average accuracy against B.
  - **Random forest:** Create a random forest of B trees with the value of B identified above. For each tree, at each terminal node, randomly consider m out of d features (m can be a value between $\sqrt{d}$ to d) for splitting. What is the average accuracy for $m = 4$?
  - Compare and comment on the classification accuracy of the original classification tree, bagged classification tree and the random forest.
- Use the inbuilt scikit learn implementation for the above experiments.

Question 2.

Regression: On Boston housing dataset (`http://lib.stat.cmu.edu/datasets/boston`), perform decision tree regression to predict the value of MEDV representing Median price of owner-occupied homes.

- Grow the regression tree using recursive binary splitting. Decision trees tend to be overly complex and do not generalize well to the data and thus, overfit. To overcome this, Stopping criteria can be used such as setting the maximum depth of the tree or minimum samples in the terminal node S. Use S= 4. Plot (Visualize) the decision tree constructed on the training data.
- **Ensemble methods to overcome overfitting:**
  - **Bagging:** for the value of S identified above, perform bagging using B bootstrapped datasets (You may consider B in the range of 50 to 500 with intervals of 50). Use cross validation to fix B. Plot average mean square error against B.
  - **Random forest:** Create a random forest of B trees with the value of B identified above. For each tree, at each terminal node, randomly consider m out of d features (m can be a value between $\sqrt{d}$ to d) for splitting. What is the average accuracy for $m = 7$?

- Compare the variance (in prediction) for the original regression tree, bagged regression tree and the random forest.
- Plot the regression curve for MEDV vs. CRIM (per capita crime rate by town), MEDV vs. INDUS (proportion of non-retail business acres per town) and MEDV vs. AGE (proportion of owner-occupied units built prior to 1940)
- Use the inbuilt scikit learn implementation for the above experiments.

Question 3.

Boosting & Comparative Analysis: Implement boosting and compare tree pruning, bagging, and boosting on both datasets.

Part A: Boosting Implementation

- **Classification (CarEvaluation):**
  - Implement AdaBoost with decision stumps (max_depth=1). Optimize the number of estimators (N) using cross-validation. Plot accuracy vs. N.
  - Visualize feature importance and compare with Random Forest's results from Question 1.
- **Regression (Boston Housing):**
  - Implement Gradient Boosting (GBRT). Tune the learning rate ($\eta \in \{0.01, 0.1, 0.5\}$) and number of estimators (N) via cross-validation.
  - Plot MSE vs. N for each η.

Part B: Comparative Analysis

- **Compare:**
  - Pruned decision tree (optimize `max_depth` or `min_samples_leaf`).
  - Bagging (with optimal B from Questions 1/2).
  - Boosting (AdaBoost/GBRT).
- **Create a table** summarizing:

Method	Test Accuracy/MSE	Training Time (s)	Prediction Variance
Pruned Tree			
Bagging			
Boosting			

- **Discussion:**
  - Which method generalizes best for each dataset?
  - When would you prefer pruning over ensemble methods? Consider factors like dataset size, noise, and interpretability.
- Use scikit-learn for all implementations.